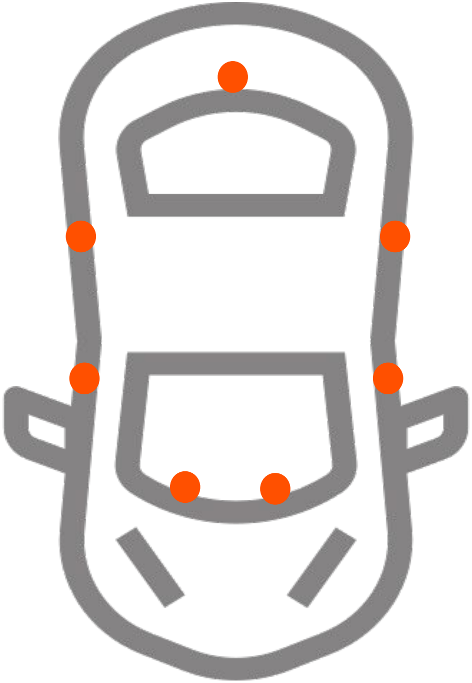
# Access Delegation with OAuth2

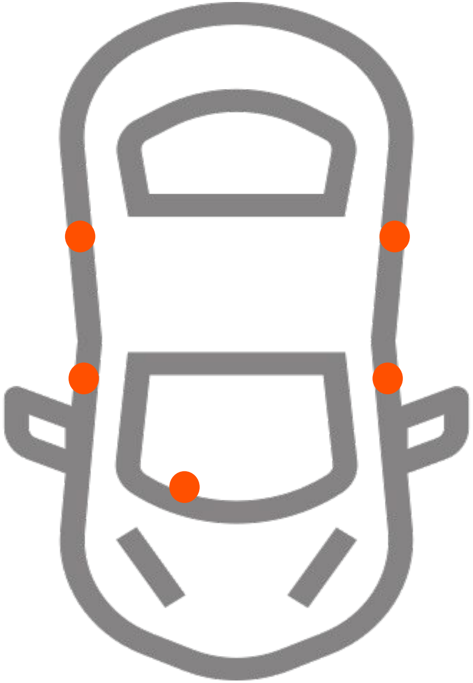
# Access Delegation

- Authorizing an application to access your resources on your behalf
- Enables granting access to execute limited actions depending on the need
- Granted permissions can be revoked from the delegates easily

# Example: Master Key vs. Valet Key

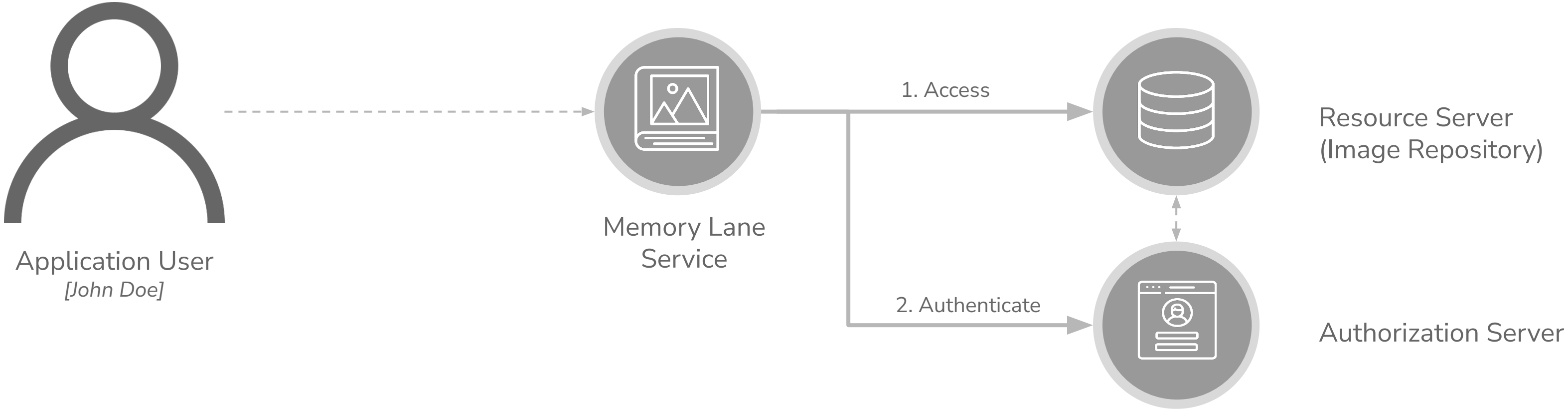


Access Points for the Master Key

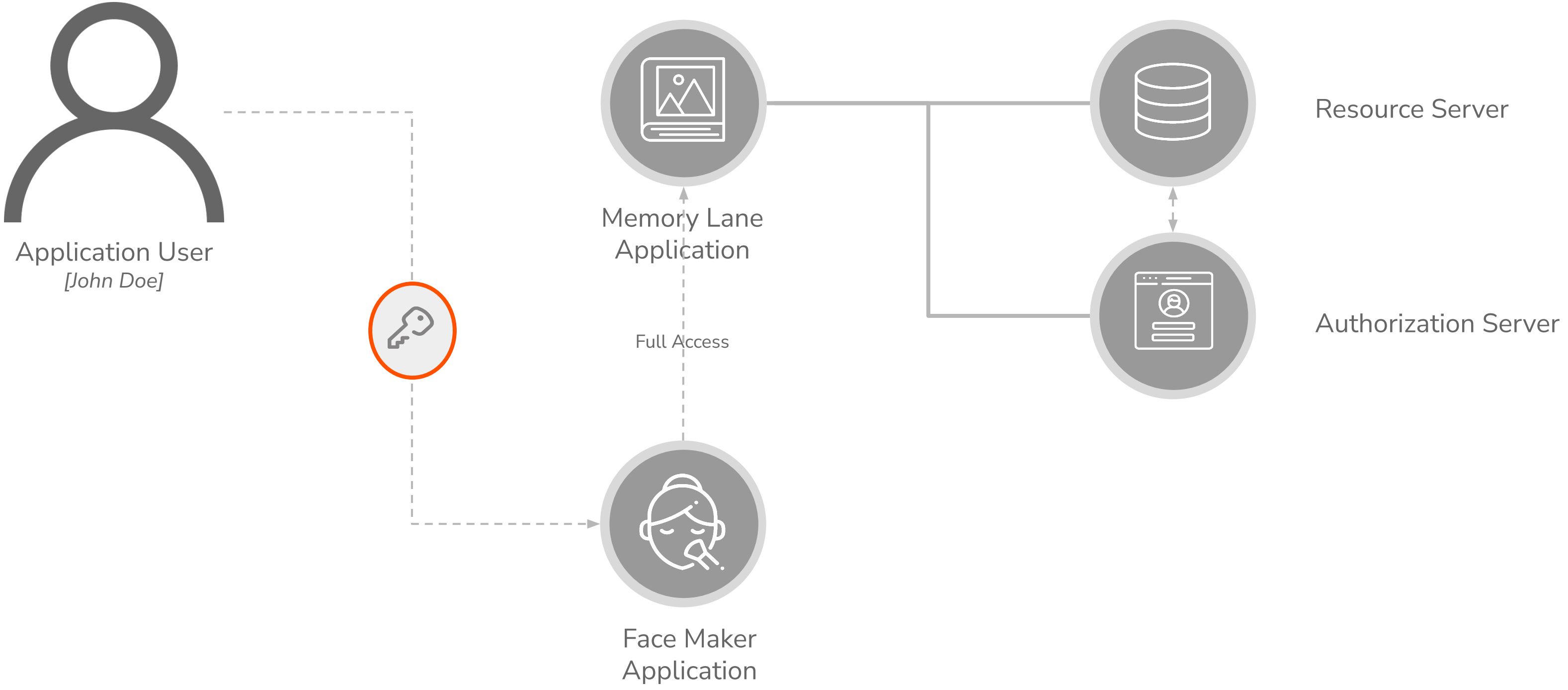


Access Points for the Valet Key

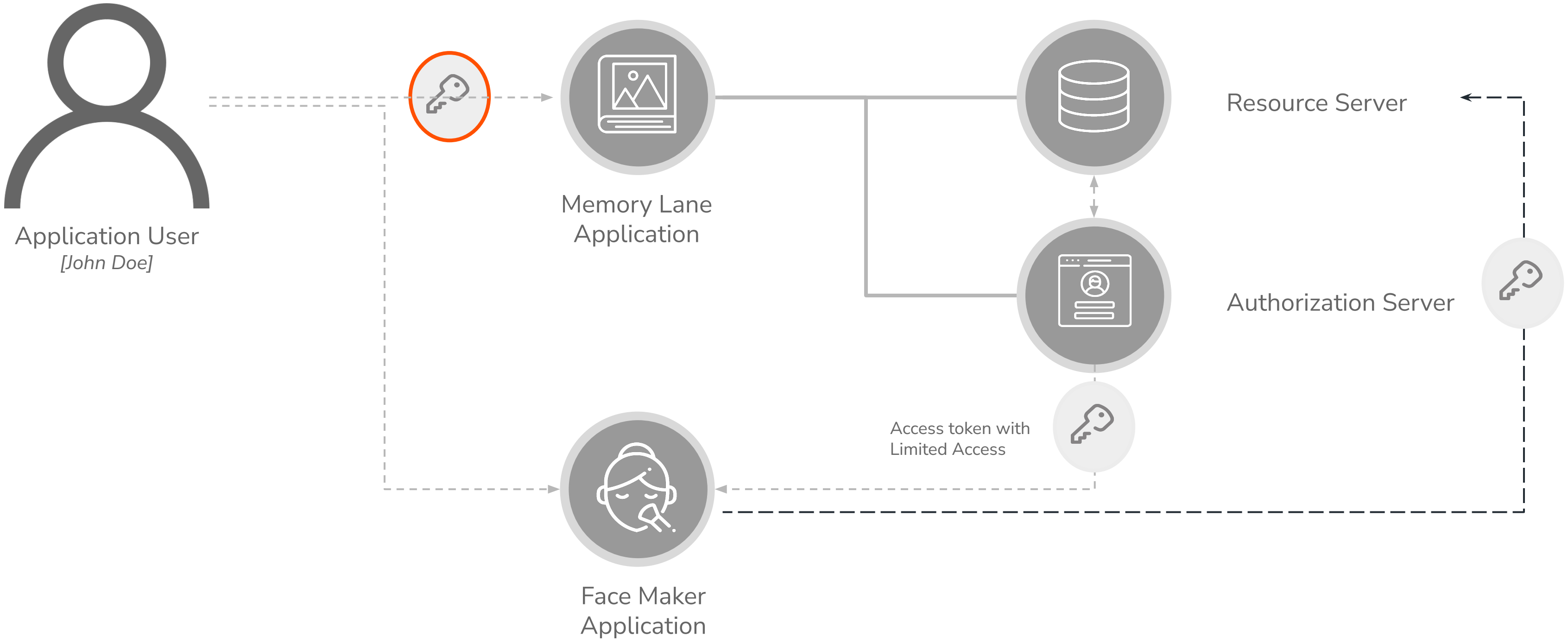
# Example: Memory Lane image service



# Example: Face Maker photo editing service



# Example: Face Maker accesses images on behalf of users



# An Overview to OAuth2

# What is OAuth2

- An industry-standard, light-weight protocol used for secure access delegation
- Enables applications to access resources without having to using the account credentials at the application site
- Generates tokens with limited access to facilitate accessing resources securely
- Extensible framework to implement new access delegation use cases

# OAuth2 Roles

## Who does what

Resource Owner

*[John Doe]*

Client Application

*[Messenger/ Face Maker]*

Authorization Server

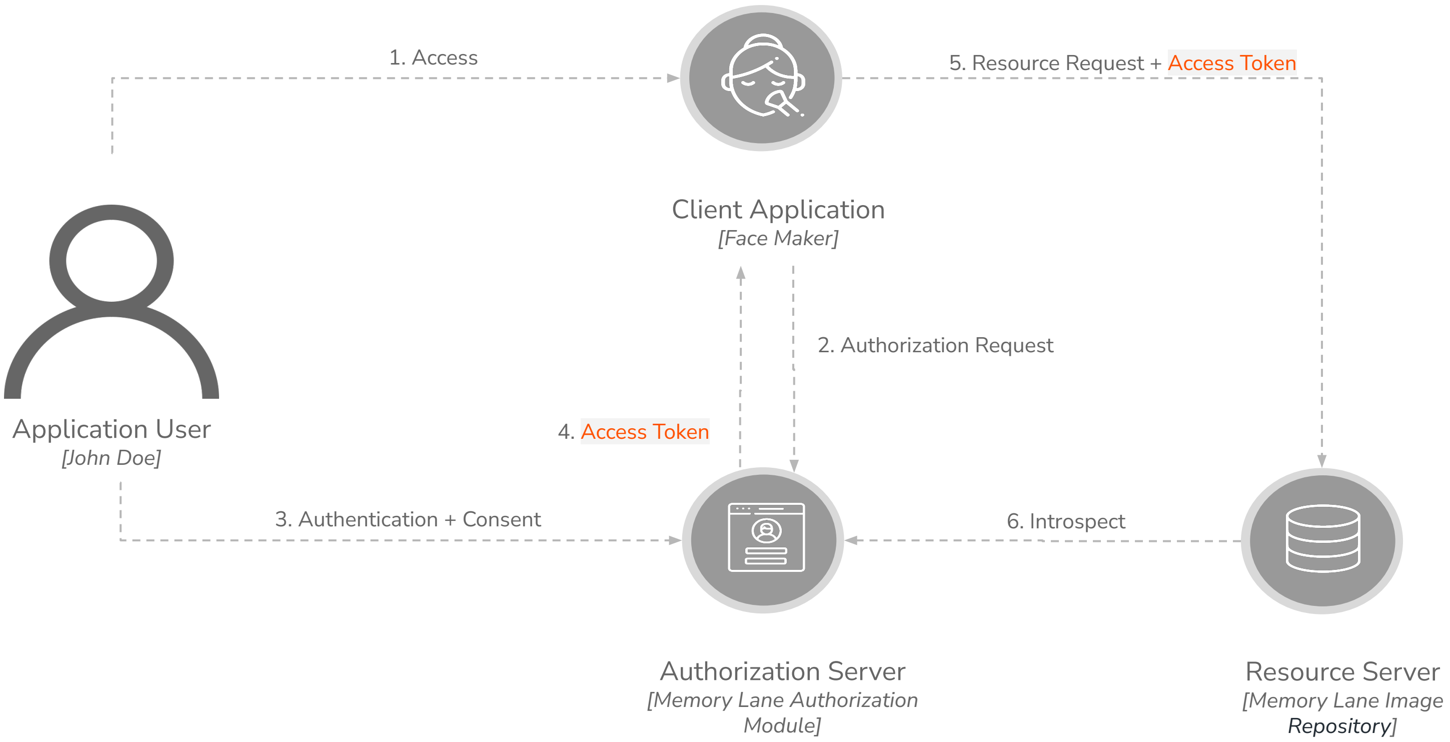
*[Google/ Facebook/ Memory Lane Authorization Module]*

Resource Server

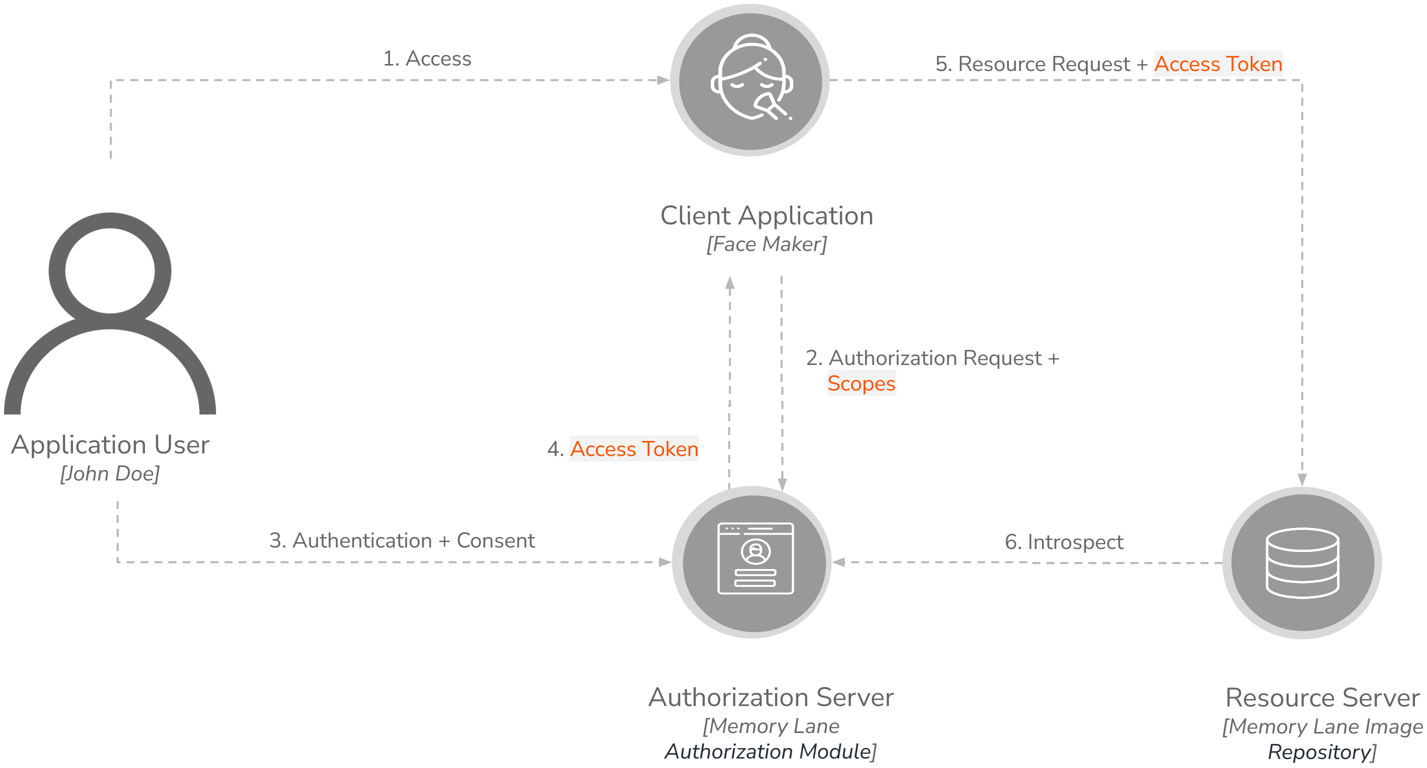
*[Memory Lane Repository/ Facebook]*

# Sample Scenarios

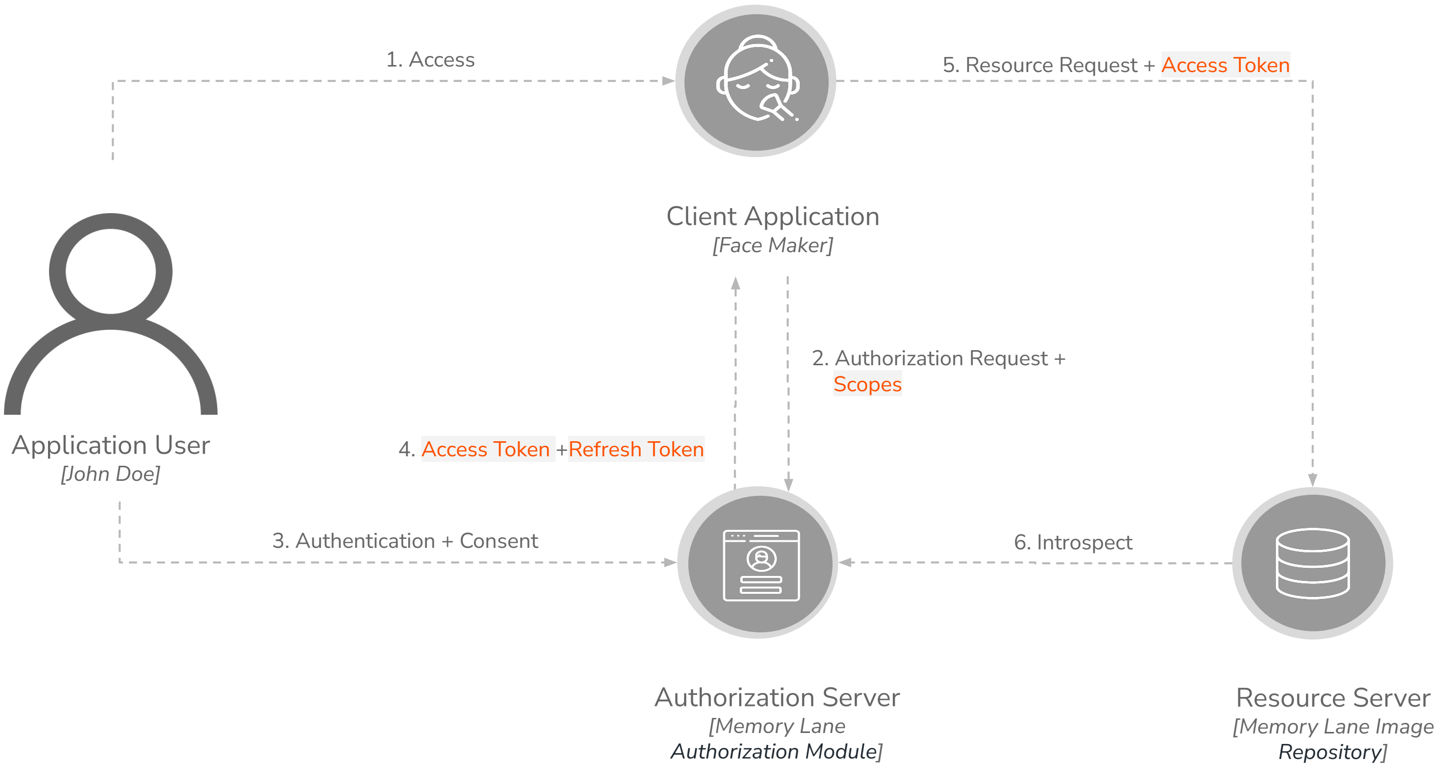
# Accessing Third-Party Resources with Access Tokens



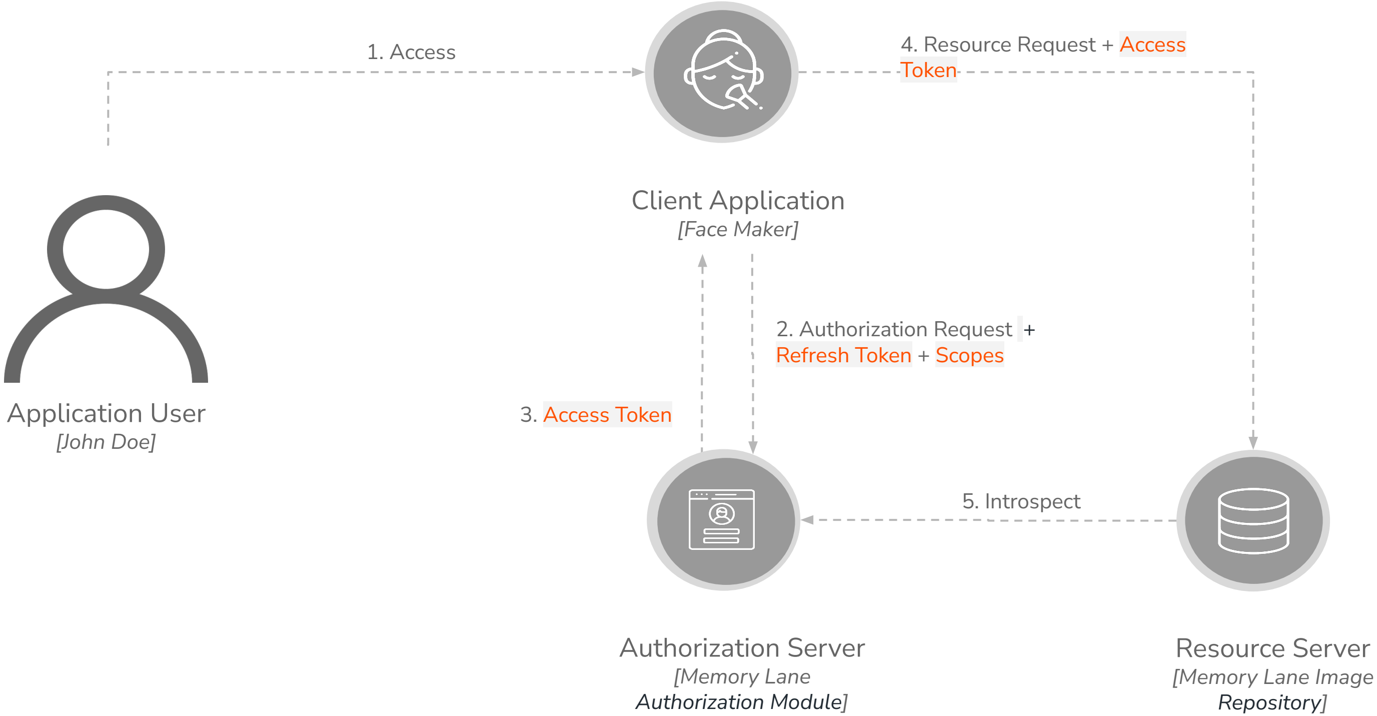
# Limiting Access to Third-Party Resources with Scopes


1. Access
5. Resource Request + Access Token
Client Application
[Face Maker]
2. Authorization Request + Scopes
4. Access Token
Application User
[John Doe]
3. Authentication + Consent
6. Introspect
Authorization Server
[Memory Lane Authorization Module]
Resource Server
[Memory Lane Image Repository]

# Sustaining the Access Token Validity



# Refreshing Access Token Upon Expiry



1. Access

Client Application
*[Face Maker]*

4. Resource Request + Access Token

2. Authorization Request + Refresh Token + Scopes

3. Access Token

Application User
*[John Doe]*

5. Introspect

Authorization Server
*[Memory Lane Authorization Module]*

Resource Server
*[Memory Lane Image Repository]*

# Tokens and Grants

# Token Types – Purpose

Access Token

**Duration**: Short
**Purpose**: Access resources

Refresh Token

**Duration**: Long
**Purpose**: Get new access tokens

# Token Types – Content

Opaque Token

**Content**: Plain text tokens

JWT Token

**Content**: Self-contained verifiable tokens

# Grant Types

- Authorization Code
- Implicit
- Password
- Client Credential
- Refresh Token
- Device Flow
- SAML2 Bearer
- JWT Bearer
- Extended

# Quick Recap

# What you learnt

1. Overview of access delegation
2. Overview of OAuth 2.0 concepts
3. Sample access delegation scenario with OAuth 2.0

# Any Questions ?

Reach us through the following channels

✉ iam-dev@wso2.org

https://stackoverflow.com/questions/tagged/wso2-identity-server

https://discord.com/invite/Xa5VubmThw

# Thanks!

wso2.com